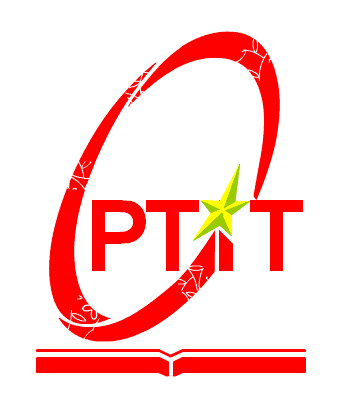
****

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**KHOA CÔNG NGHỆ THÔNG TIN 1**

-----🙞🙜🕮🙞🙜-----



**BÁO CÁO BÀI TẬP LỚN**

**“Hệ Thống Theo Dõi Thông Tin Sức Khỏe Bệnh Nhân”**

**Giảng viên hướng dẫn**: **DƯƠNG TRẦN ĐỨC**

**MÔN HỌC: IOT VÀ ỨNG DỤNG**

**NHÓM HỌC PHẦN:06**

**NHÓM BÀI TẬP LỚN:30**

**1. Lê Phan Thanh Bình - B20DCCN093**

**2. Phạm Văn Lực – B20DCCN411**

# LỜI CẢM ƠN

Lời đầu tiên, chúng em xin gửi lời cảm ơn đến Học viện Công nghệ Bưu chính Viễn Thông đã tạo điều kiện cho chúng em được học môn IoT và ứng dụng. Đặc biệt, chúng em xin gửi lời cảm ơn chân thành và sâu sắc nhất tới thầy **DƯƠNG TRẦN ĐỨC**, giảng viên bộ môn đã hướng dẫn và truyền đạt những kiến thức hết sức bổ ích và quý báu trong suốt thời gian học tập vừa qua.

Với vốn hiểu biết sâu rộng và kinh nghiệm nhiều năm giảng dạy cũng như làm việc trong môi trường công nghệ thông tin, thầy khiến chúng em thật sự ấn tượng, khâm phục và “ngỡ ngàng” trước những hiểu biết của thầy. Những kiến thức, kinh nghiệm thầy chia sẻ khiến chúng em thật sự ngưỡng mộ và một phần hãnh diện khi được học dưới sự hướng dẫn của thầy.

IoT và ứng dụng là một môn học nền tảng với một lượng kiến thức tương đối nhiều và cần đầu tư thời gian học khá lớn nên khó có thể hiểu rõ, hiểu sâu nhanh chóng khi thời lượng học trên lớp có hạn. Chính vì vậy mặc dù thầy đã truyền đạt rất nhiệt tình và tận tâm nhưng do năng lực có hạn, khả năng tư duy và khả năng tiếp thu còn chưa nhanh nên trong báo cáo bài tập lớn này chúng em khó có thể tránh khỏi được những sai sót. Do đó, chúng em kính mong thầy xem xét và bổ sung giúp đỡ chúng em hoàn thiện bài báo cáo này một cách đầy đủ và đúng đắn hơn.

Một lần nữa, chúng em xin chân thành cảm ơn thầy!

|  |  |
| --- | --- |
|  |  |

Mục lục

[LỜI CẢM ƠN 2](#_Toc153823051)

[A, Mô tả về dự án: 4](#_Toc153823052)

[B, Mô tả cơ bản về chức năng 4](#_Toc153823053)

[C, Các thiết bị. 4](#_Toc153823054)

[D, Đóng góp của các thành viên trong nhóm. 5](#_Toc153823055)

[E, Bên trong hệ thống: 5](#_Toc153823056)

[1. Mạng neural chẩn đoán bệnh 5](#_Toc153823057)

[2. Backend dùng để cung cấp api cho esp và lưu thông tin vào database 9](#_Toc153823058)

[3. Code giao diện của bác sĩ và giao tiếp với phụ tá thông qua hivemq 10](#_Toc153823059)

[4. Code giao diện của phụ tá và giao tiếp với bác sĩ thông qua hivemq 21](#_Toc153823060)

[5. Code Arduino 24](#_Toc153823061)

[F, Các màn hình demo chức năng: 25](#_Toc153823062)

[1. Màn hình bác sĩ hiển thị các thông số và trạng thái bệnh nhân: 26](#_Toc153823063)

[2. Màn hình bác sĩ thêm dữ liệu vào database: 26](#_Toc153823064)

[3.Màn hình bác sĩ yêu cầu phụ tá: 27](#_Toc153823065)

[4.Màn hình phụ tá khi chưa được bác sĩ yêu cầu: 27](#_Toc153823066)

[5.Màn hình phụ tá khi được bác sĩ yêu cầu: 28](#_Toc153823067)

[6.Màn hình bác sĩ sau khi phụ tá phản hồi: 28](#_Toc153823068)

# A, Mô tả về dự án:

\*Đề bài: Hệ thống theo dõi thông tin sức khỏe bệnh nhân

\*Hệ thống theo dõi thông tin về nhiệt độ, độ ẩm môi trường xung quanh bệnh nhân và nhịp tim, nồng độ oxy trong máu của bệnh nhân. Hệ thống sẽ thu nhận các dữ liệu đó, phân tích và dự đoán tình trạng bệnh nhân hiện tại. Sau đó các thông số và tình trạng bệnh nhân sẽ hiển thị cho bác sĩ thấy. Nếu bác sĩ thấy trường hợp đó cần sự giúp đỡ từ phụ tá thì bác sĩ sẽ gọi phụ tá và phía phụ tá sẽ nhận được thông báo gọi từ bác sĩ. Phía phụ tá có thể chấp nhận hoặc từ chối yêu cầu từ bác sĩ. Sau đó bác sĩ cũng sẽ nhận được phản hồi từ phụ tá là chấp nhận hay từ chối.

# B, Mô tả cơ bản về chức năng

1. Xem thông tin về nhiệt độ, độ ẩm của môi trường xung quanh.

2. Xem thông tin về nhịp tim và nồng độ oxy trong máu.

3. Từ 4 thông số trên sẽ phân tích và dự đoán về tình trạng bệnh nhân dựa trên mạng neural.

4. Lưu thông thông tin về nhiệt độ, độ ẩm, nhịp tim, nồng độ oxy trong máu và trạng thái bệnh nhân vào db để lấy dữ liệu train lại AI giúp dự đoán về tình trạng bệnh nhân tốt hơn.

5. Tạo sự liên lạc giữa bác sĩ và phụ tá. Để khi bác sĩ cần gọi phụ tá và phụ tá cũng phản hồi ngược lại được cho bác sĩ.

# C, Các thiết bị.

1. Esp32.

2. Cảm biến nhiệt độ, độ ẩm DHT22.

3. Cảm biến đo nồng độ oxy trong máu, nhịp tim MAX30100.

# D, Đóng góp của các thành viên trong nhóm.

1.Phạm Văn Lực-B20DCCN411(Nhóm trưởng)

-Code backend phần giao tiếp với database

-Code frontend phần bác sĩ

-Code arduino phần kết nối với backend để lấy kết quả phân tích dự đoán trạng thái bệnh nhân và trả kết quả cho frontend để hiển thị

2.Lê Phan Thanh Bình-B20DCCN093

-Code backend phần phân tích dữ liệu từ cảm biến và đưa ra dự đoán trạng thái bệnh nhân

-Code frontend phần phụ tá

-Code Arduino phần kết nối với cảm biến và nhận các thông số từ cảm biến trả về

-Lắp ráp để kết nối esp32 với cảm biến dht22 và max30100.

# E, Bên trong hệ thống:

### 1. Mạng neural chẩn đoán bệnh

Forward propagation

*𝑍*[1]=*𝑊*[1]*𝑋*+*𝑏*[1]

*𝐴*[1]=*𝑔*ReLU(*𝑍*[1]))

*𝑍*[2]=*𝑊*[2]*𝐴*[1]+*𝑏*[2]

*𝐴*[2]=*𝑔*softmax(*𝑍*[2])

Backward propagation

*𝑑𝑍*[2]=*𝐴*[2]−*𝑌*

*𝑑𝑊*[2]=(1/*𝑚)𝑑𝑍*[2]*𝐴*[1]*(chuyển vị)*

*𝑑𝐵*[2]=(1/*𝑚)*Σ*𝑑𝑍*[2]

*𝑑𝑍*[1]=*𝑊*[2]*(chuyển vị). 𝑑𝑍*[2]∗*𝑔*[1]′(*𝑧*[1])

*𝑑𝑊*[1]=(1/*𝑚)𝑑𝑍*[1]*𝐴*[0]*(chuyển vị)*

*𝑑𝐵*[1]=(1/*𝑚)*Σ*𝑑𝑍*[1]

Parameter updates

*𝑊*[2]:=*𝑊*[2]−*𝛼𝑑𝑊*[2]

*B*[2]:=*B*[2]−*𝛼𝑑B*[2]

*𝑊*[1]:=*𝑊*[1]−*𝛼𝑑𝑊*[1]

*B*[1]:=*B*[1]−*𝛼𝑑B*[1]

**Kích cỡ ma trận:**

Forward propagation

*𝐴*[0]=*𝑋*: 4 x m (m là số bản ghi trong database)

*𝑍*[1]∼*𝐴*[1]: 30 x m

*𝑊*[1]: 30 x 4 (*𝑊*[1]*𝐴*[0]∼*𝑍*[1])

*𝐵*[1]: 30 x 1

*𝑍*[2]∼*𝐴*[2]: (số bệnh trong database) x m

*𝑊*[2]: (số bệnh trong database) x 30 (*𝑊*[2]*𝐴*[1]∼*𝑍*[2])

*𝐵*[2]: (số bệnh trong database) x 1

Backprop

*𝑑𝑍*[2]: số bệnh trong database x m ( *𝐴*[2])

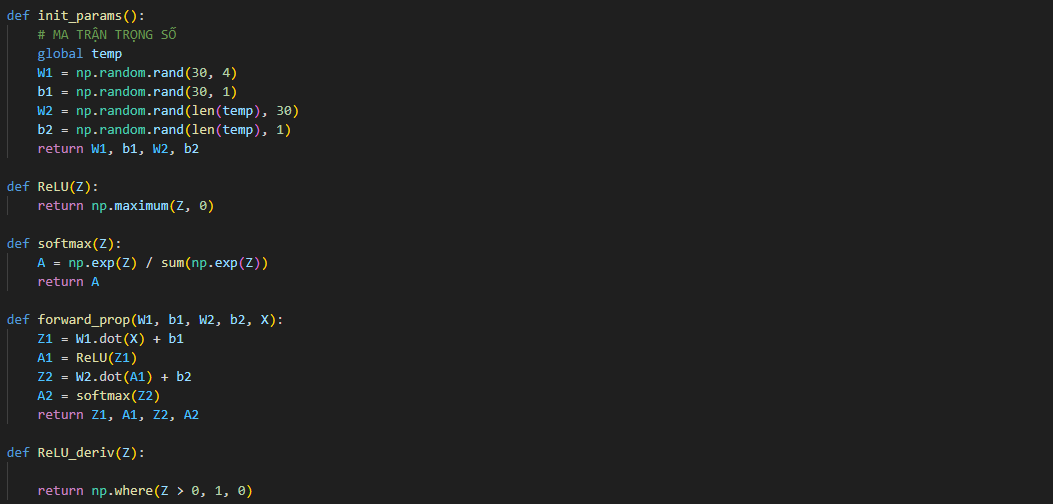
*𝑑𝑊*[2]:số bệnh trong database x 30

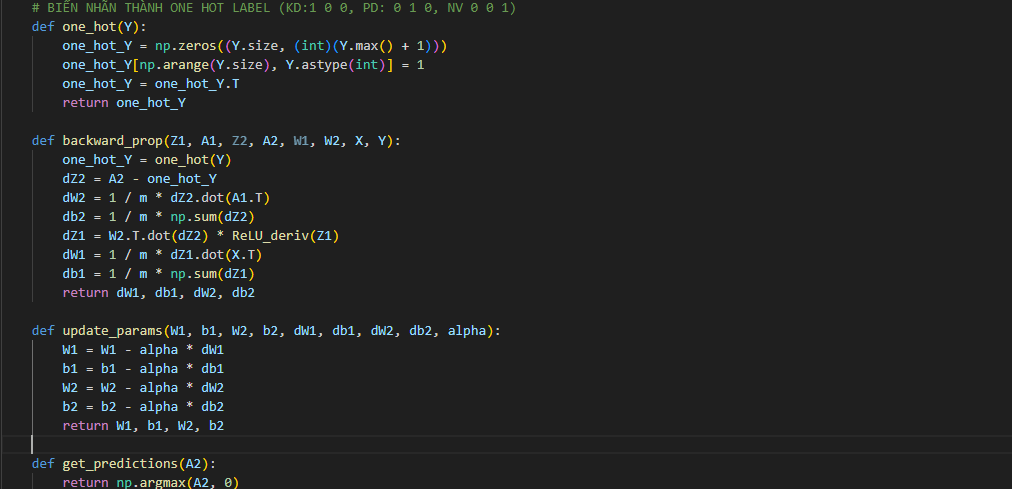
*𝑑𝐵*[2]: số bệnh trong database x 1

*𝑑𝑍*[1]: 30 x m ( *𝐴*[1])

*𝑑𝑊*[1]: 30 x 4

*𝑑𝐵*[1]: 30 x 1

****

****

****

**-init\_params: khởi tạo ma trận trọng số và bias.**

**-ReLU, softmax: định nghĩa hàm kích hoạt relu và softmax**

**-forward\_prop: lan truyền xuôi**

**-ReLU\_deriv: định nghĩa đạo hàm hàm kích hoạt relu**

**-one\_hot: one hot encoding nhãn bệnh thành ma trận one hot**

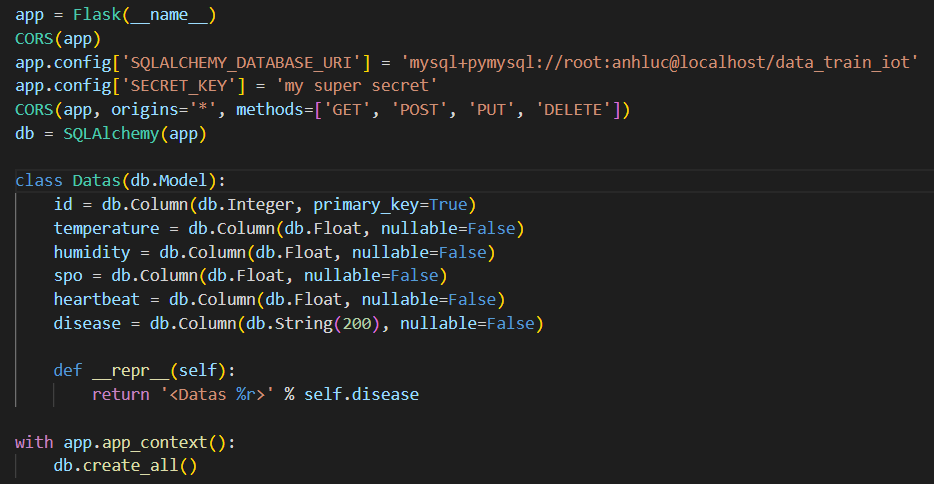
**-backward\_prop: lan truyền ngược**

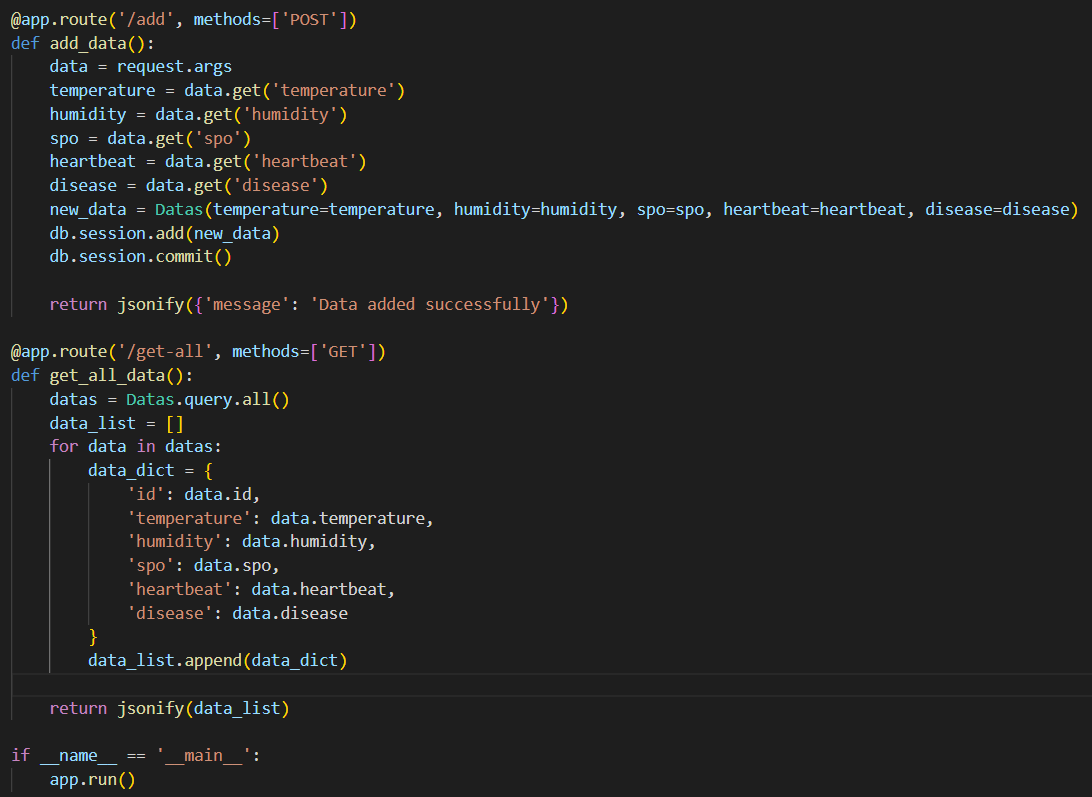
**-update\_params: cập nhật lại trọng số mỗi iteration**

**-gradient\_descent: thuật toán gradient descent để tối ưu các giá trị trọng số và bias**

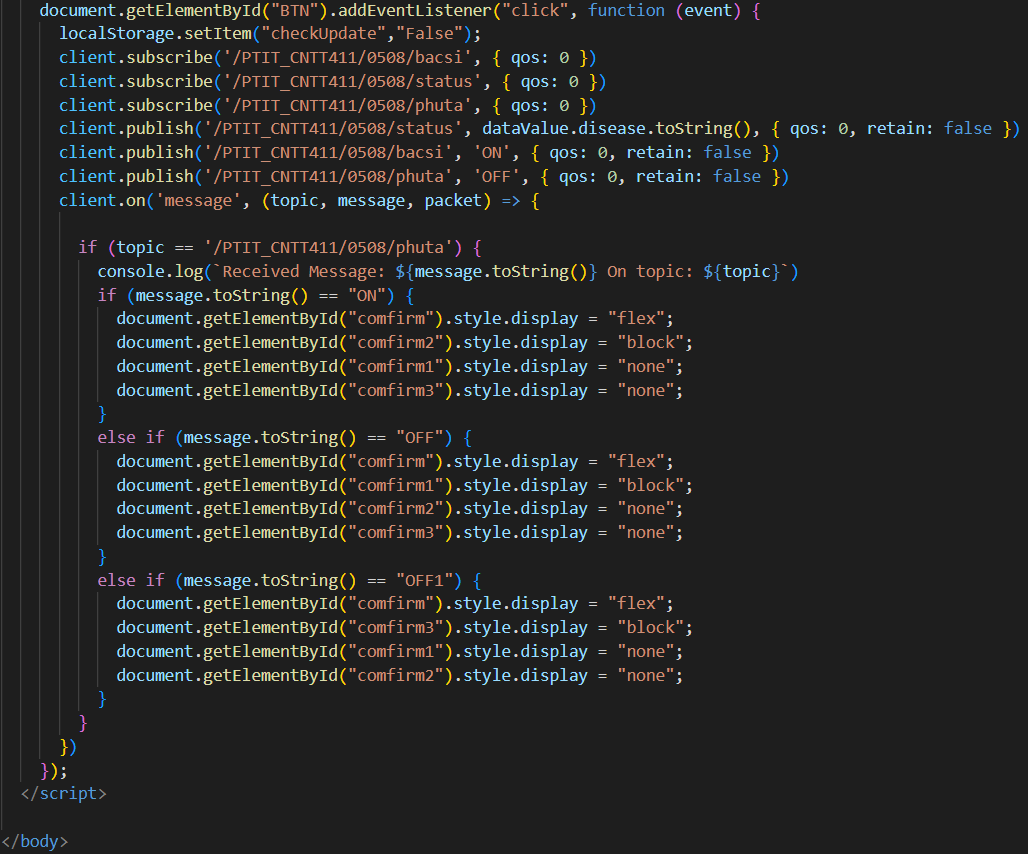
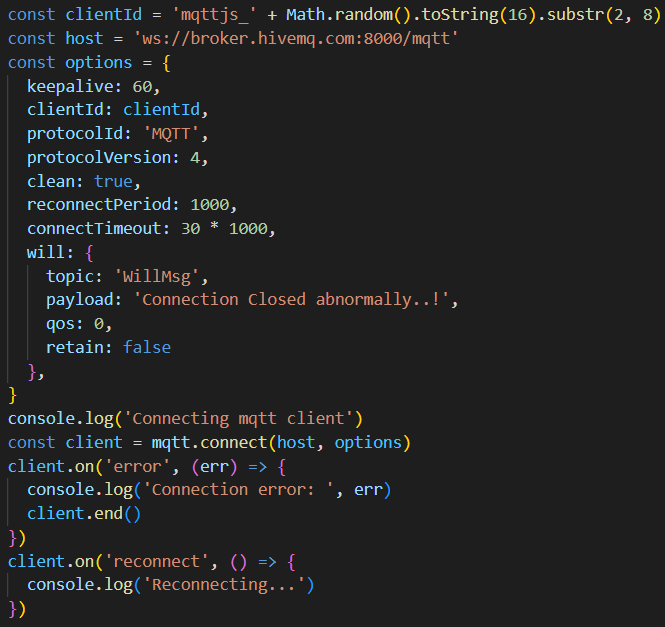
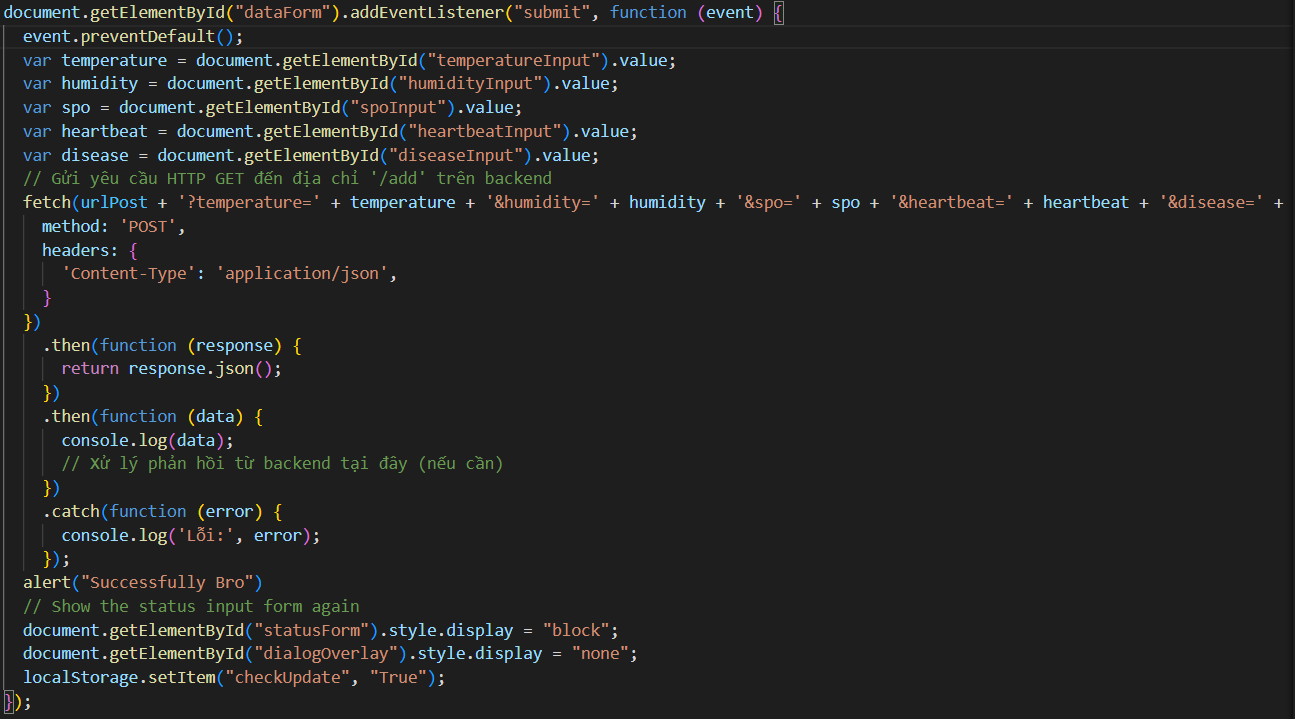
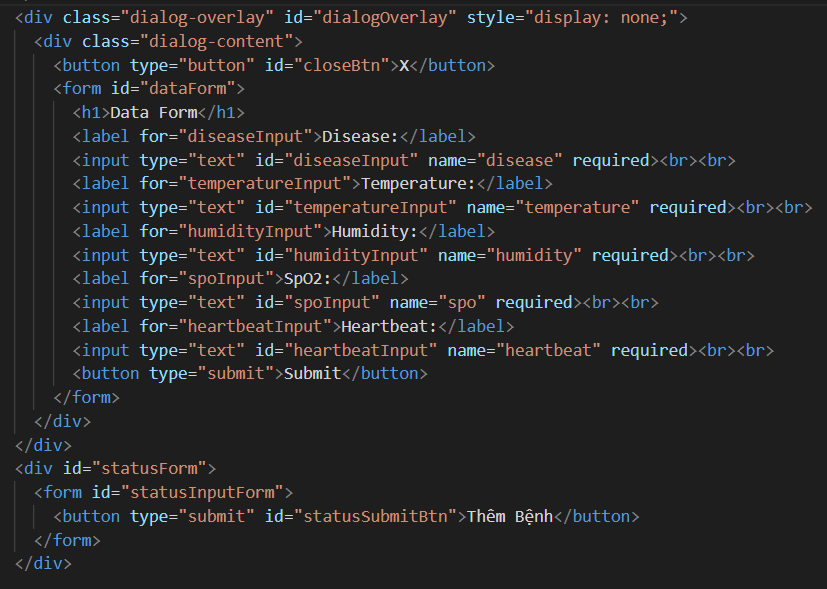
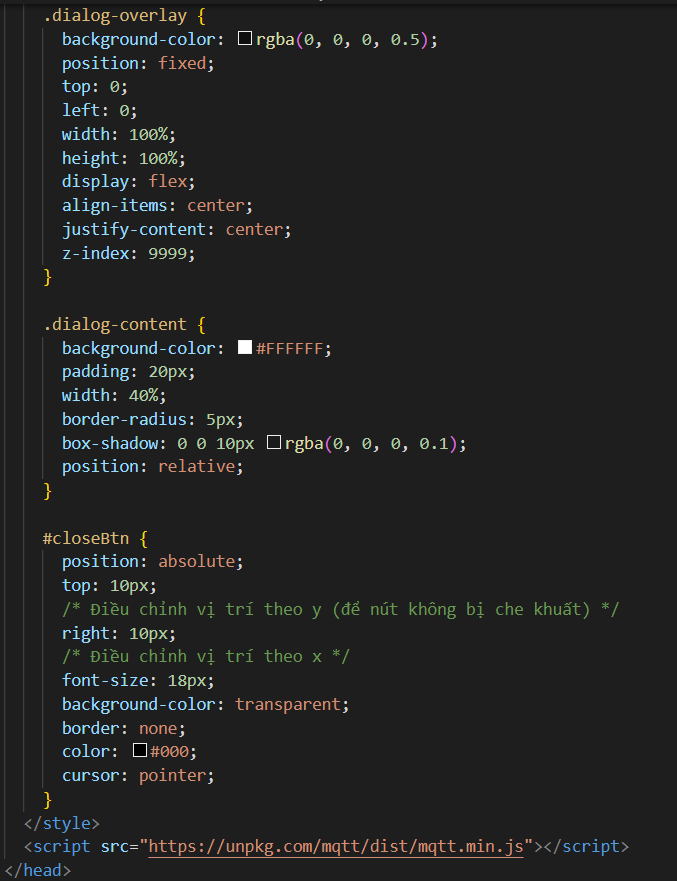
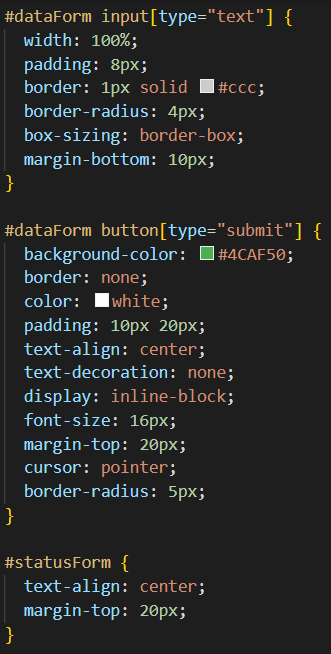
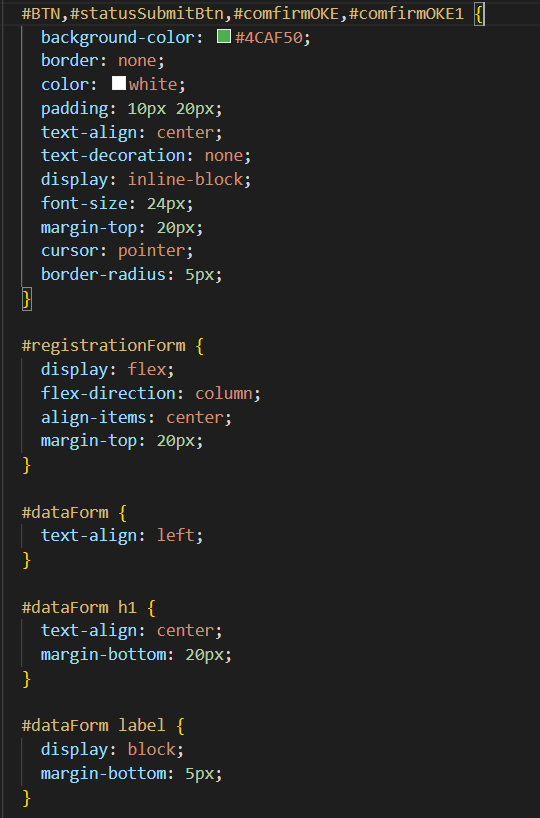
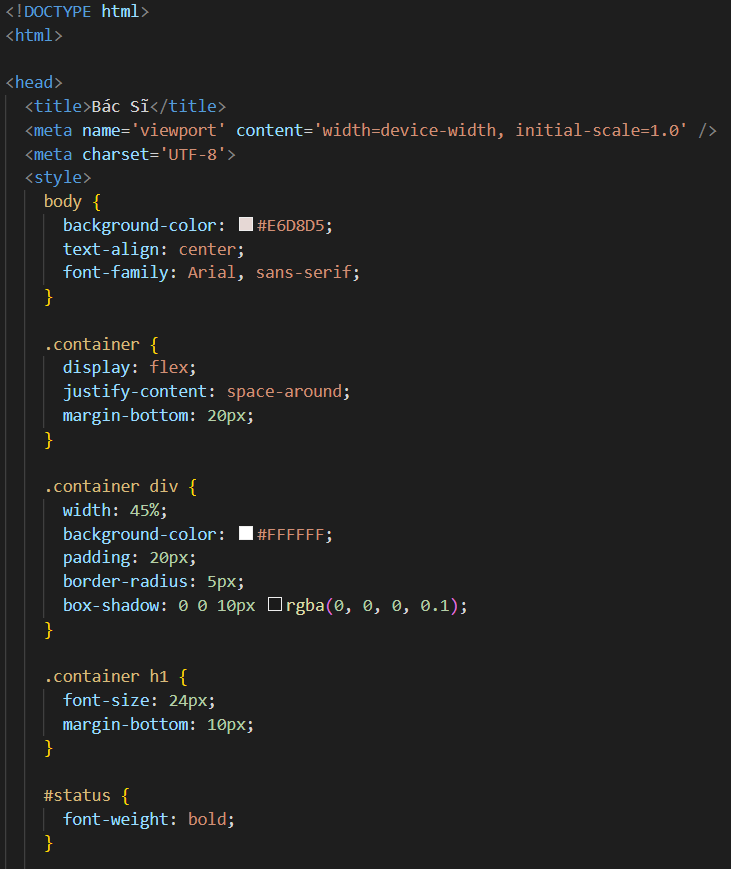
**-getDisease: hàm dự đoán bệnh dựa trên thông số các cảm biến gửi về**

### 2. Backend dùng để cung cấp api cho esp và lưu thông tin vào database

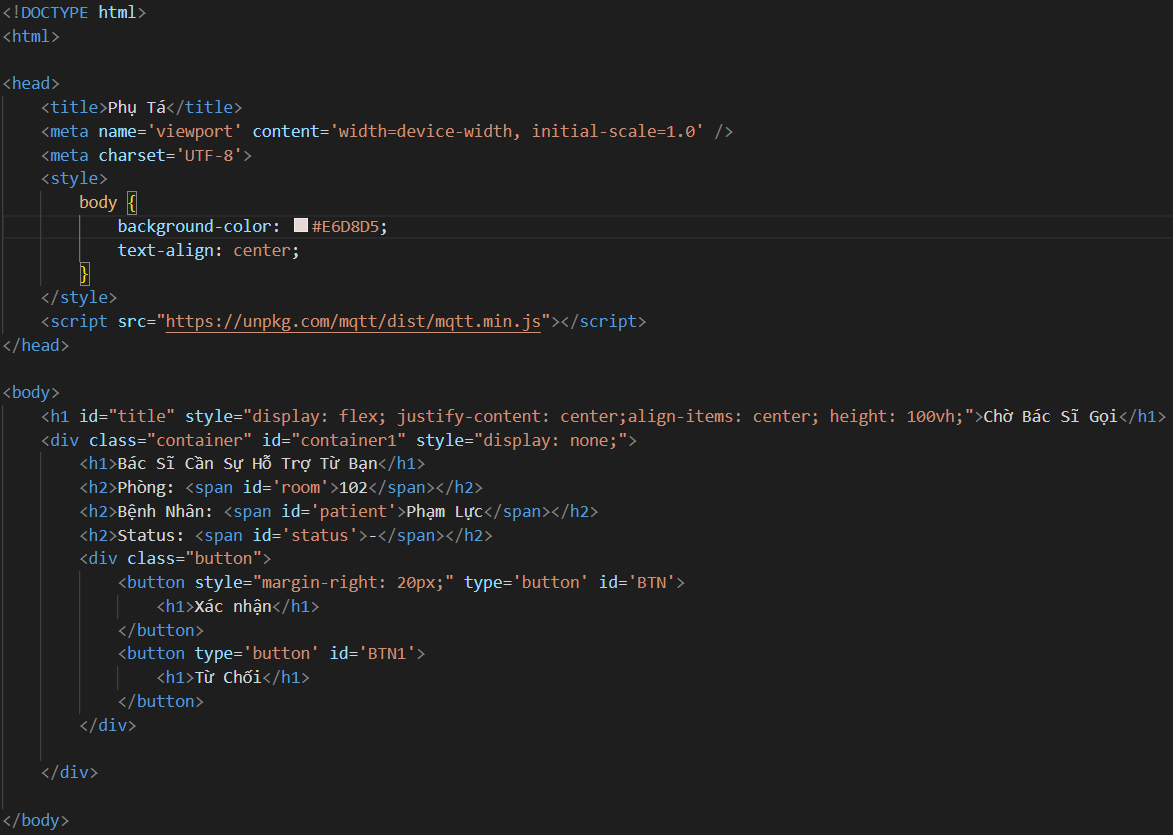
****

****

### 3. Code giao diện của bác sĩ và giao tiếp với phụ tá thông qua hivemq

****

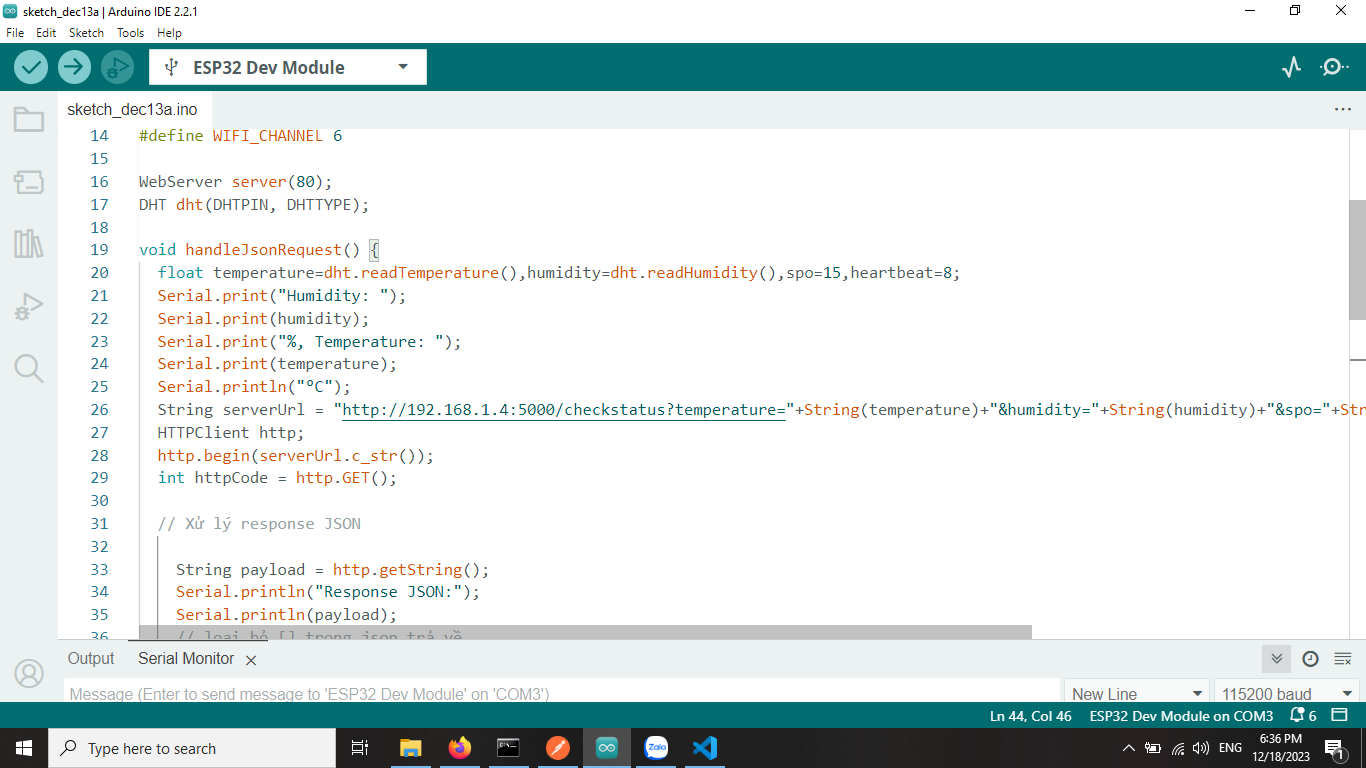
### 4. Code giao diện của phụ tá và giao tiếp với bác sĩ thông qua hivemq

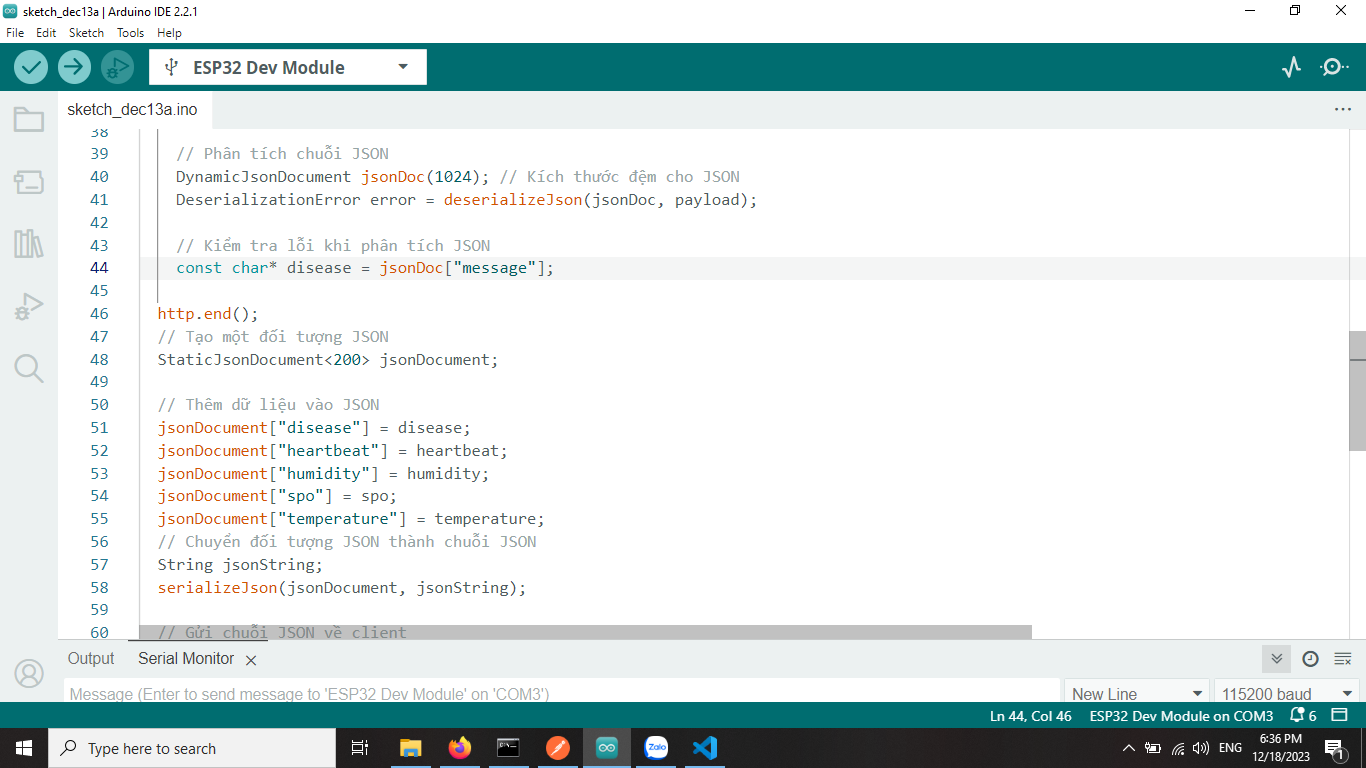
****

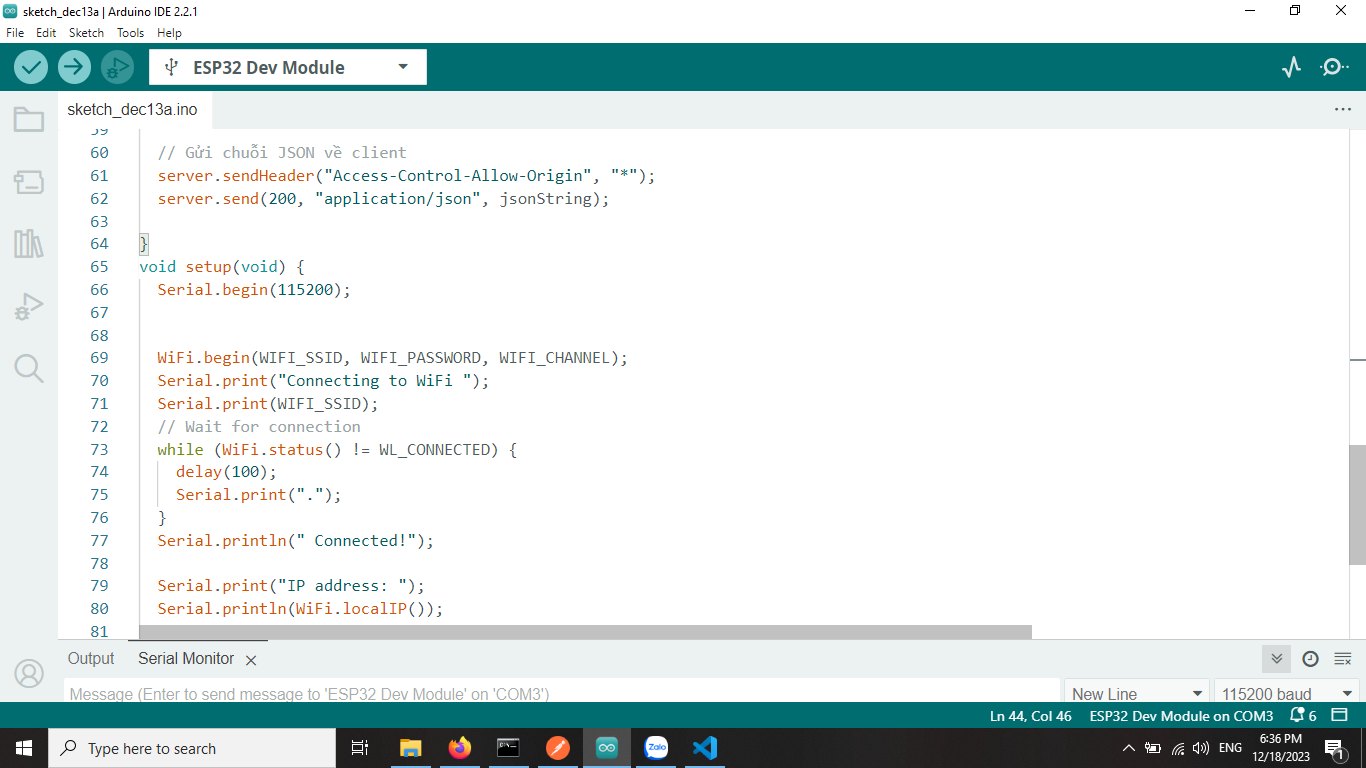
****

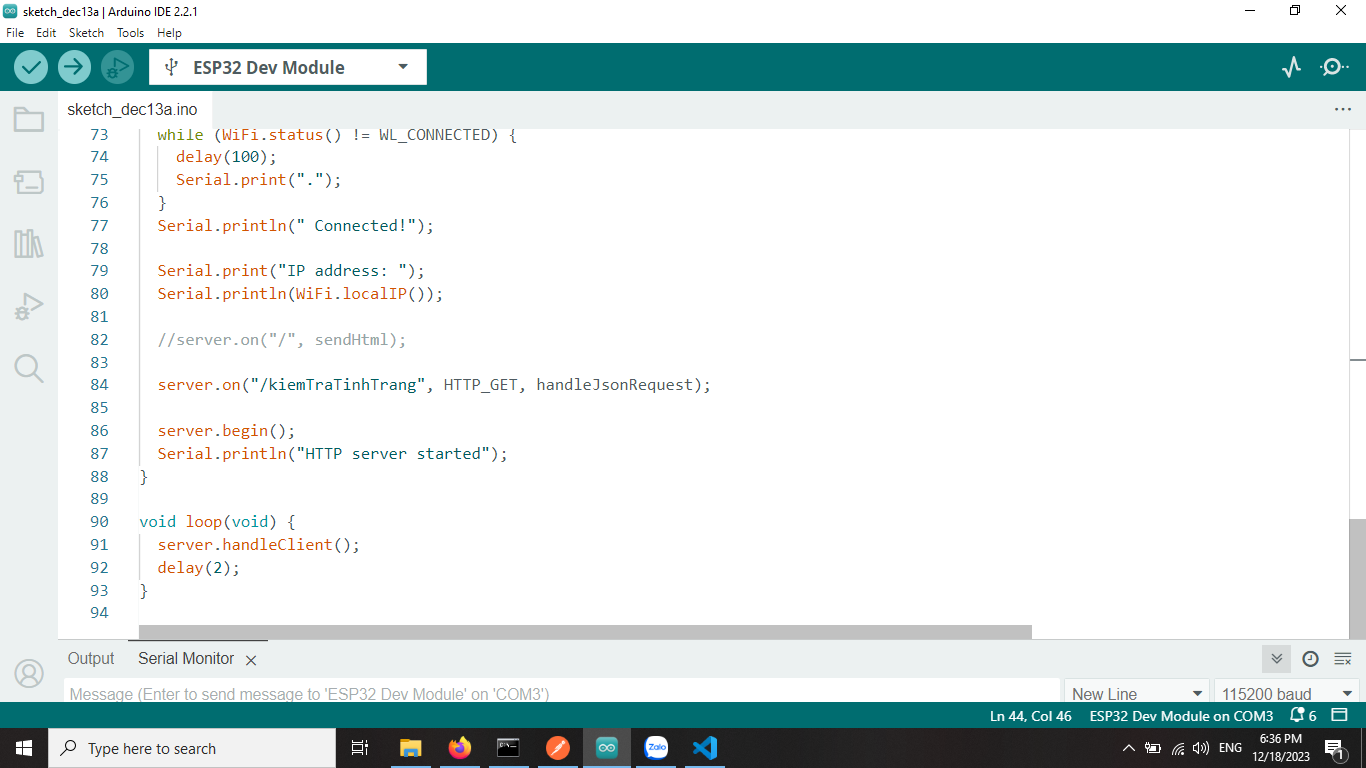
****

### 5. Code Arduino

****

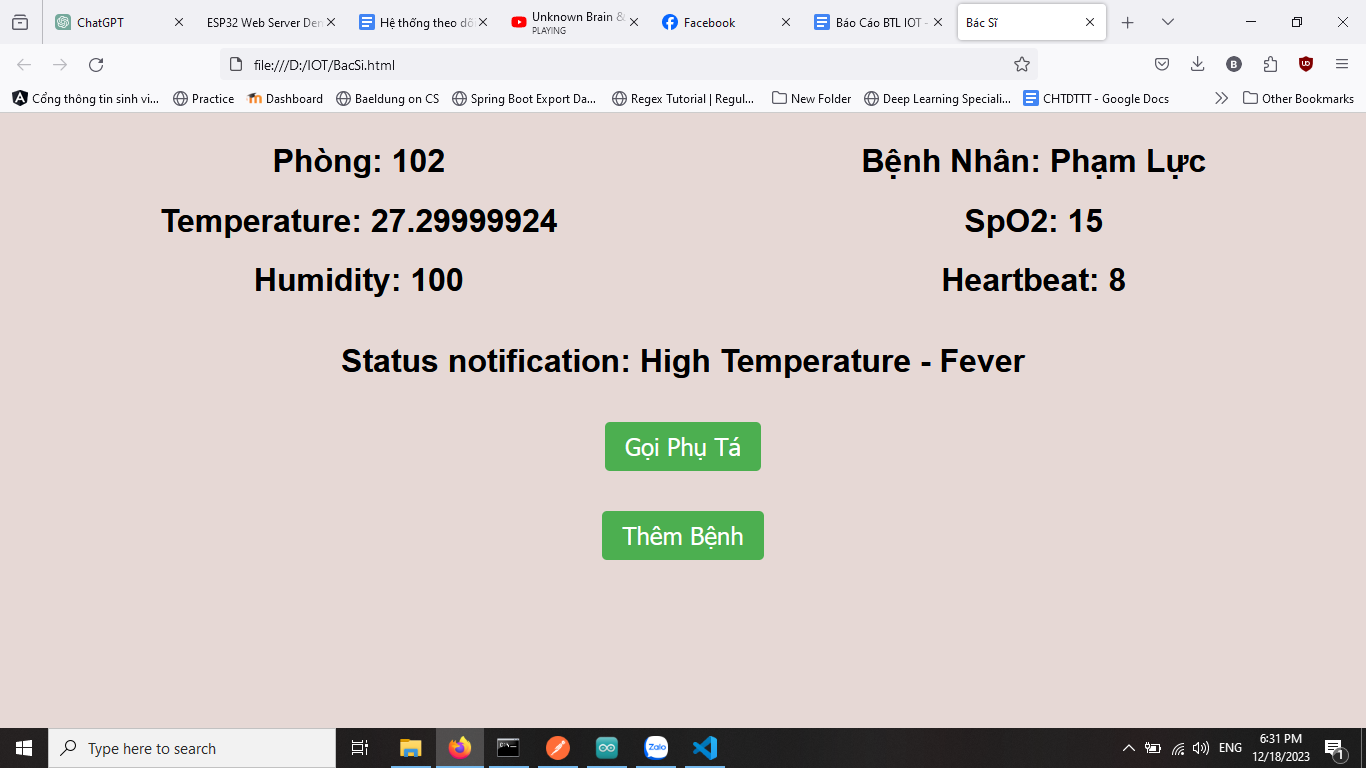
****

****

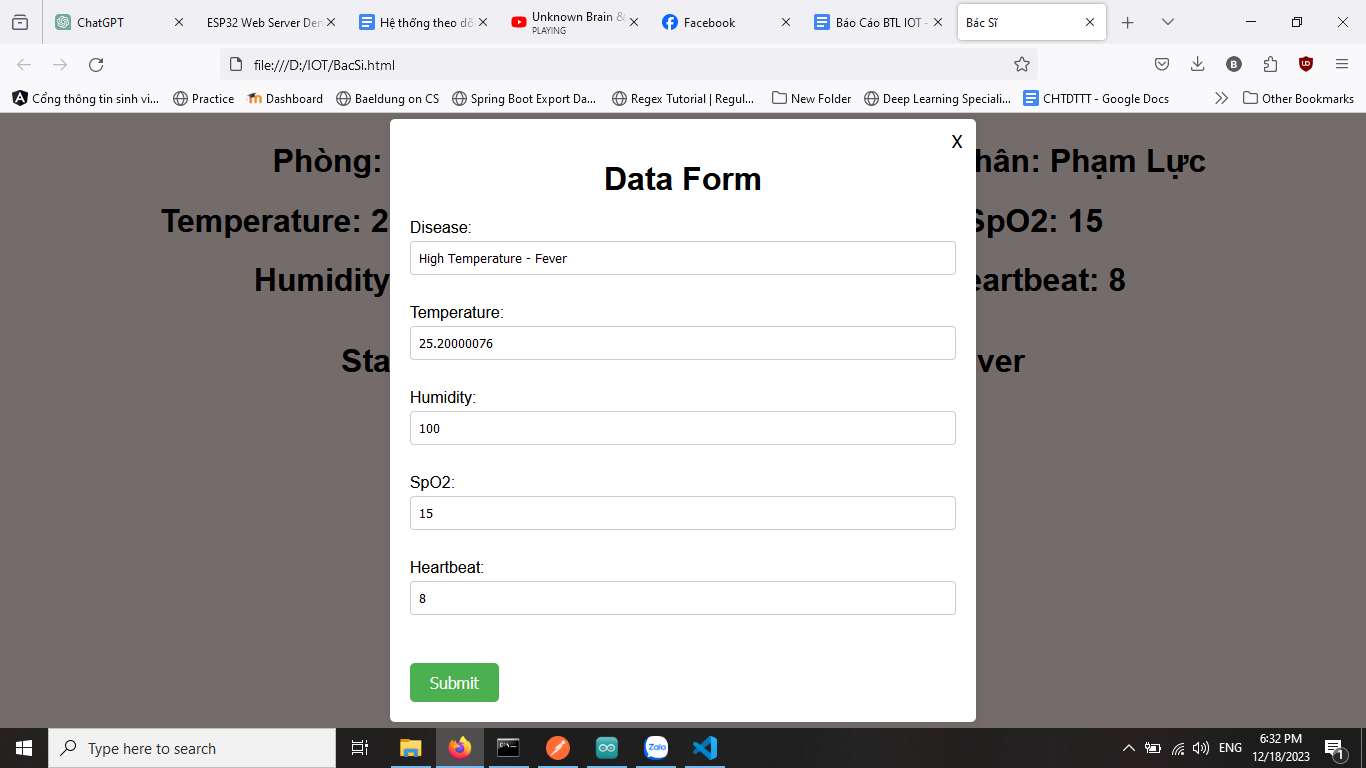
****

# F, Các màn hình demo chức năng:

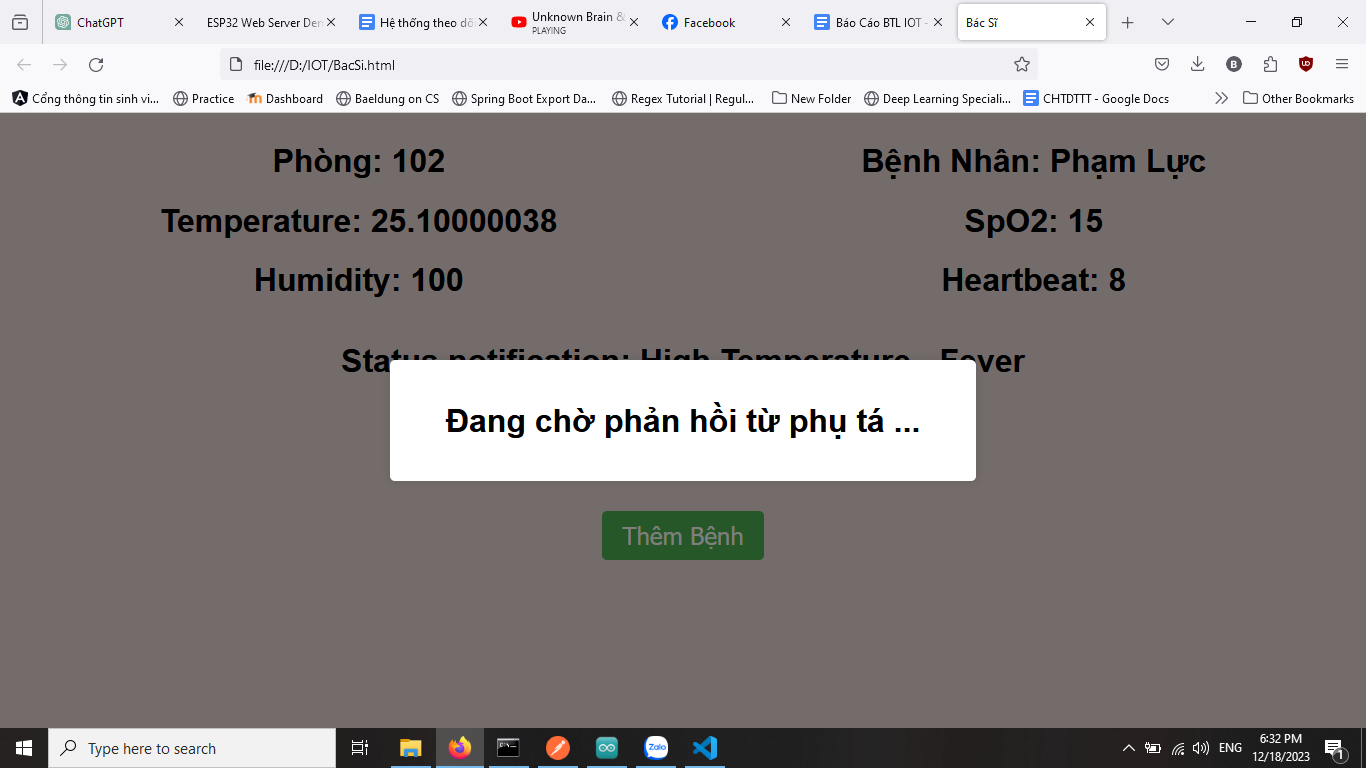
### 1. Màn hình bác sĩ hiển thị các thông số và trạng thái bệnh nhân:



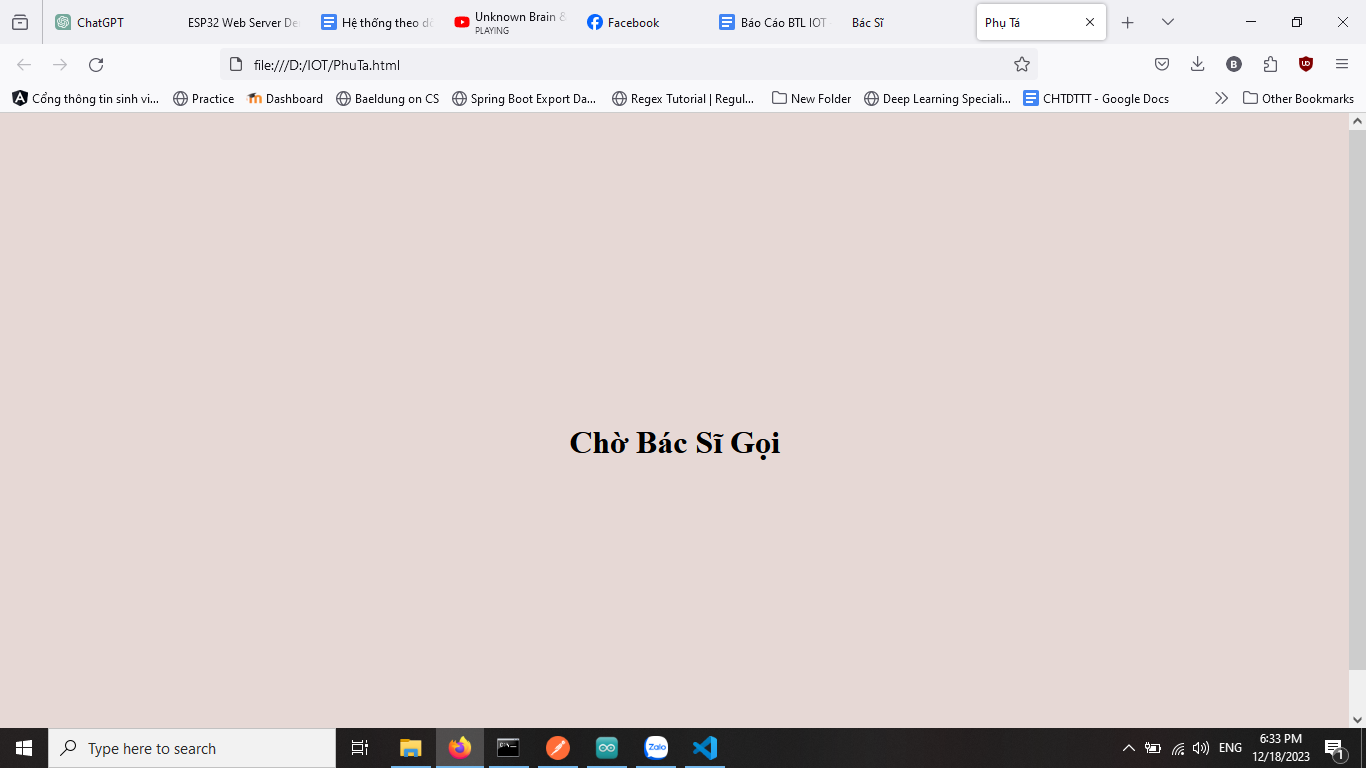
### 2. Màn hình bác sĩ thêm dữ liệu vào database:



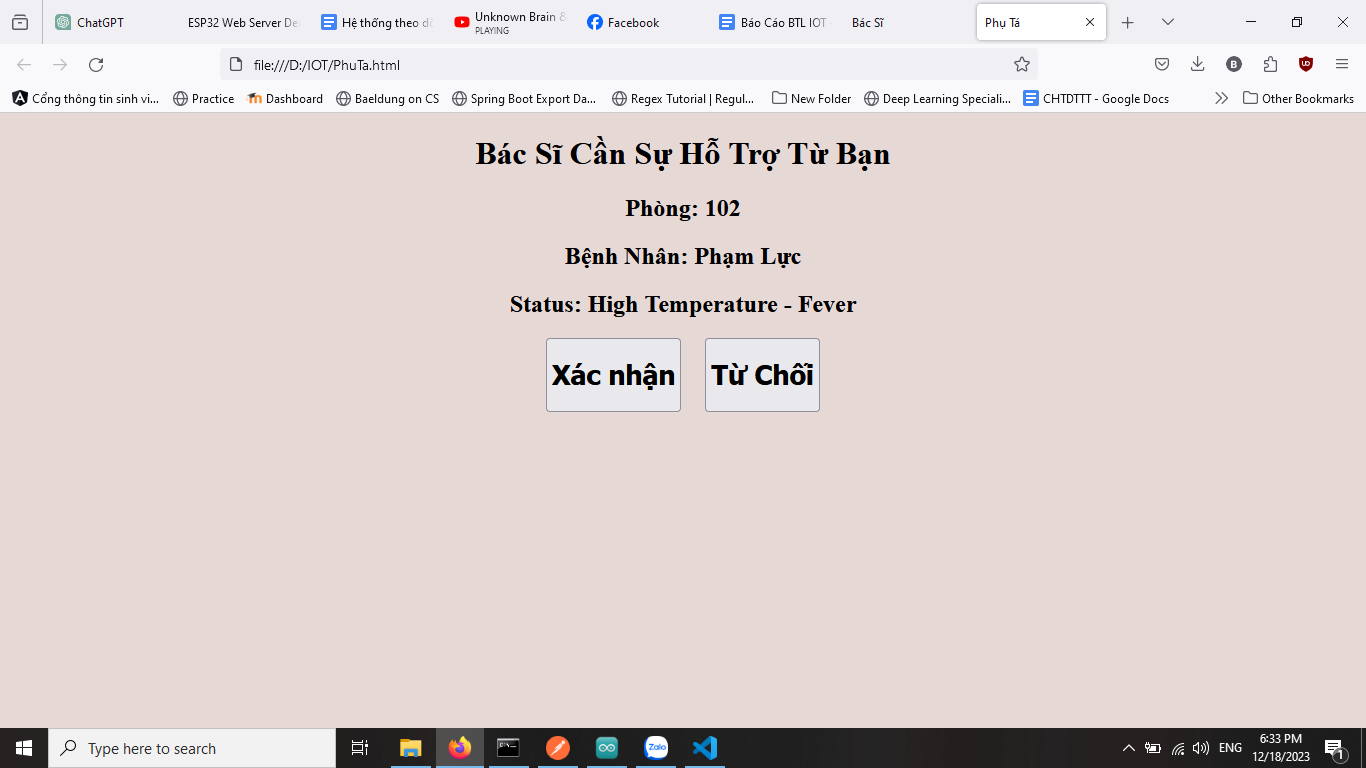
### 3.Màn hình bác sĩ yêu cầu phụ tá:



### 4.Màn hình phụ tá khi chưa được bác sĩ yêu cầu:

****

### 5.Màn hình phụ tá khi được bác sĩ yêu cầu:



### 6.Màn hình bác sĩ sau khi phụ tá phản hồi:

